IN THE CLAIMS

1. (Currently amended) A method of serving data to a plurality of clients in a client-server environment, comprising the steps of:

~~providing~~ generating a plurality of versions of given data in which at least two versions of the given data have different overheads associated therewith, the overhead of a given version of the given data comprising a quantity of processing resources required to serve the given version of the given data;

assigning individual clients to one of a plurality of quality-of-service classes; and

satisfying requests so that a client belonging to a high quality-of-service class is given preferential access to data versions which require higher overheads to serve while a client belonging to a low quality-of-service class receives a data version which requires lower overhead to serve.

2. (Original) The method of claim 1, wherein the overhead to serve a version is correlated with a quality of the version.

3. (Currently amended) The method of claim 2, wherein the plurality of versions comprise images of different resolutions and clients belonging to [[a]] the high quality-of-service class are given preferential access to higher resolution images while a client belonging to the low quality-of-service class receives a lower resolution image.

4. (Original) The method of claim 2, wherein the quality of a version is correlated with a processing time required to create the version.

5. (Original) The method of claim 1, wherein the overhead to serve a version is correlated with how current the version is.

6. (Original) The method of claim 1, further comprising the step of:

in response to a system load exceeding a threshold, satisfying a higher percentage of requests from clients belonging to a lower quality-of-service class with a version requiring lower overhead to serve.

7. (Original) The method of claim 1, wherein the server comprises multiple nodes and different nodes provide data versions requiring different overheads to serve.

8. (Original) The method of claim 1, further comprising the step of implementing a quality-of-service policy that specifies at least one of content quality and latency.

9. (Original) The method of claim 8, wherein one or more clients belonging to a premium service class are served with high content quality and low latency.

10. (Original) The method of claim 8, wherein one or more clients belonging to a medium service class are served with one of high content quality and low latency.

11. (Original) The method of claim 8, wherein one or more clients belonging to a best-effort service class are served with unspecified content quality and latency.

12. (Original) The method of claim 1, wherein a client request is routed using at least one of an identity of the client, a quality of content, a load on at least one server, a data distribution on at least one server, and a capacity of at least one server.

13. (Original) The method of claim 1, wherein a client is assigned to a quality-of-service class by program logic that is externalized from the server.

14. (Canceled)

15. (Original) The method of claim 1, further comprising the step of satisfying requests using a policy determined by program logic that is externalized from the server.

16. (Canceled)

17. (Currently amended) Apparatus for serving data to a plurality of clients in a client-server environment, comprising:

a memory, and

at least one processor coupled to the memory and operative to: (i) ~~provide~~ generate a plurality of versions of given data in which at least two versions of the given data have different overheads associated therewith, the overhead of a given version of the given data comprising a quantity of processing resources required to serve the given version of the given data; (ii) assign individual clients to one of a plurality of quality-of-service classes; and (iii) satisfy requests so that a client belonging to a high quality-of-service class is given preferential access to data versions which require higher overheads to serve while a client belonging to a low quality-of-service class receives a data version which requires lower overhead to serve.

18. (Original) The apparatus of claim 17, wherein the overhead to serve a version is correlated with a quality of the version.

19. (Currently amended) The apparatus of claim 18, wherein the plurality of versions comprise images of different resolutions and clients belonging to [[a]] the high quality-of-service class are given preferential access to higher resolution images while a client belonging to the low quality-of-service class receives a lower resolution image.

20. (Original) The apparatus of claim 18, wherein the quality of a version is correlated with a processing time required to create the version.

21. (Original) The apparatus of claim 17, wherein the overhead to serve a version is correlated with how current the version is.

22. (Original) The apparatus of claim 17, wherein the at least one processor is further operative to, in response to a system load exceeding a threshold, satisfy a higher percentage of requests from clients belonging to a lower quality-of-service class with a version requiring lower overhead to serve.

23. (Original) The apparatus of claim 17, wherein the at least one processor comprises multiple nodes and different nodes provide data versions requiring different overheads to serve.

24. (Original) The apparatus of claim 17, wherein the at least one processor is further operative to implement a quality-of-service policy that specifies at least one of content quality and latency.

25. (Original) The apparatus of claim 24, wherein one or more clients belonging to a premium service class are served with high content quality and low latency.

26. (Original) The apparatus of claim 24, wherein one or more clients belonging to a medium service class are served with one of high content quality and low latency.

27. (Original) The apparatus of claim 24, wherein one or more clients belonging to a best-effort service class are served with unspecified content quality and latency.

28. (Original) The apparatus of claim 17, wherein a client request is routed using at least one of an identity of the client, a quality of content, a load on at least one server, a data distribution on at least one server, and a capacity of at least one server.

29. (Currently amended) An article of manufacture for use in serving data to a plurality of clients in a client-server environment, comprising a machine readable storage medium containing one or more programs which when executed implement the steps of:

~~providing~~ generating a plurality of versions of given data in which at least two versions of the given data have different overheads associated therewith, the overhead of a given version of the given data comprising a quantity of processing resources required to serve the given version of the given data;

assigning individual clients to one of a plurality of quality-of-service classes; and

satisfying requests so that a client belonging to a high quality-of-service class is given preferential access to data versions which require higher overheads to serve while a client belonging to a low quality-of-service class receives a data version which requires lower overhead to serve.

30. (Currently amended) A system, comprising:

a plurality of clients, each client belonging to a quality-of-service class;

a load balancer for sending requests from clients to at least one back-end server; and

at least one back-end server for ~~providing~~ generating versions of objects in which at least two versions of a given object have different overheads associated therewith, the overhead of a given version of the given data comprising a quantity of processing resources required to serve the given version of the given data.

31. (Currently amended) A method of providing a data serving service, comprising the step of:

a service provider: (i) ~~providing~~ generating a plurality of versions of given data in which at least two versions of the given data have different overheads associated therewith, the overhead of a given version of the given data comprising a quantity of processing resources required to serve the given version of the given data; (ii) assigning individual clients to one of a plurality of quality-of-service classes; and (iii) satisfying requests so that a client belonging to a high quality-of-service class is given preferential access to data versions which require higher overheads to serve while a client belonging to a low quality-of-service class receives a data version which requires lower overhead to serve.

32. (Original) The method of claim 31, wherein the data serving service comprises a quality-of-service policy specification.

33. (Original) The method of claim 32, wherein the quality-of-service policy specification comprises:

a plurality of subscriptions, each subscription being specified by content quality and service latency, wherein a limited premium service subscription is served with high content quality in low service latency, a medium service subscription is served with a high content quality or a low service latency, and an unlimited best-effort service subscription is served with unspecified content quality and latency.

34. (Original) The method of claim 31, wherein the service provider modifies data content and how the data content is served to clients in response to one or more changing conditions.

35. (Original) The method of claim 34, wherein one or more changing conditions comprises a source of a bottleneck.

36. (Original) The method of claim 31, wherein the step of assigning individual clients to one of a plurality of quality-of-service classes is based on a client payment.

37. (Currently amended) A method of serving data to a plurality of clients, comprising the steps of:

establishing at least two quality-of-service classes; and

satisfying requests so that a client belonging to one quality-of-service class is served with one version of given data having one overhead associated therewith, while a client belonging to another quality-of-service class is served with another version of the given data having another overhead associated therewith, the overhead of a given version of the given data comprising a quantity of processing resources required to serve the given version of the given data.

38. (New) The system of claim 30, where the at least one back-end server comprises:
at least a first back-end server for generating a first version of the given object; and
at least a second back-end server for generating a second version of the given object;
wherein the first and second versions of the given object have different overheads associated therewith.

39. (New) The method of claim 37, wherein the one version of the given data is served by one back-end server while the other version of the given data is served by another back-end server.